

# 基于FPGA的空域精细可扩展编码预处理器

乔保军<sup>1,2</sup>, 石峰<sup>1</sup>

(1. 北京理工大学计算机科学与工程系, 北京 100081; 2. 河南大学计算机与信息工程学院, 开封 475001)

**摘要:** FPGA 越来越多地应用于各种数字信号处理系统中。针对空域精细可扩展编码算法, 提出一种基于FPGA的预处理器设计方案, 该编码预处理器占用较少的外部存储空间缓存帧数据, 充分利用数据操作之间的并行性和流水性, 生成视频数据的多描述码流。FPGA 实现结果表明, 该预处理器能满足应用空域精细可扩展编码算法的视频压缩传输系统的功能要求和实时性要求。

**关键词:** FPGA; 精细可扩展编码; 预处理器

## Preprocessor for Spatial Fine Granularity Scalable Coding Based on FPGA

QIAO Baojun<sup>1,2</sup>, SHI Feng<sup>1</sup>

(1. Department of Computer Science and Engineering, Beijing Institute of Technology, Beijing 100081;

2. College of Computer and Information Engineering, Henan University, Kaifeng 475001)

**【Abstract】** FPGA is more and more used in many kinds digital signal processing systems. A preprocessor for S-FGS based on FPGA is presented which can produce five description coding streams of video images. The preprocessor does not use very large RAM to store frame data, and can fully apply the parallel and pipeline of data operation. Experimental results show the design can meet the function and real-time requirement of S-FGS.

**【Key words】** FPGA; Fine granularity scalable coding; Preprocessor

FPGA具有可编程、资源丰富、开发周期短、成本低等优点, 它已经成为数字电路研究开发的一种重要实现形式, 并广泛应用于各种数字信号处理系统当中<sup>[1]</sup>。网络接入系统的异构和网络带宽的变化是网络中传输视频图像遇到的主要问题之一。为了提高对异构和带宽的适应性, 各种视频编码方法得到越来越多的研究与应用: 精细可扩展编码(Fine Granular Scalability, FGS)<sup>[2]</sup>, 分层编码(Layered Coding, LC)<sup>[3]</sup>, 多描述编码(Multiple Description Coding, MDC)<sup>[4]</sup>。为了使空域分层也具有类似FGS的精细可扩展性, 文献[5]提出了空域精细可扩展编码算法(S-FGS), 该算法产生细粒度的可扩展码流, 使压缩码流具备灵活的网络带宽适应性。这种算法的视频编码系统中, 最重要的一个部分就是视频数据的多描述码流的生成。

本文提出一种基于FPGA的空域精细可扩展视频编码预处理器的设计和实现方案, 该预处理器占用较少的外部存储空间缓存帧数据, 充分利用数据操作之间的并行性和流水性, 生成视频数据的多描述码流, 能满足应用空域精细可扩展编码算法的视频压缩传输系统的功能要求和实时性要求。

### 1 空域精细可扩展编码系统

S-FGS 算法首先对原始图像下采样, 生成满足应用要求的最低分辨率图像作为基本层, 然后从基本层逐渐插值和预测重建图像, 将逐次预测的残差数据编码成可以在任意点截断的增强层码流。

利用FPGA和DSP构建空域精细可扩展编码系统, 其中用FPGA实现编码预处理器和相应的编码控制, 用双DSP实现多个描述的压缩编码, 一个DSP用于基本层的编码, 另一个编码增强层。系统采用Philips公司的视频A/D芯片SAA7114H进行视频采集; DSP采用的是TI公司的

TMS320C6205; FPGA采用Xilinx公司的50万门级芯片XC2V500。该视频编码系统的硬件结构如图1所示。

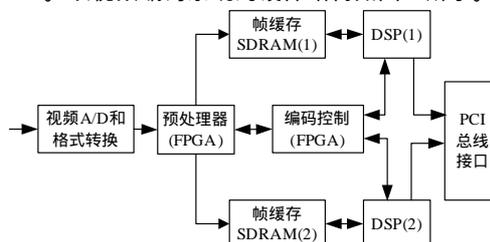


图1 空域精细可扩展编码系统结构原理

预处理器负责对视频数据进行多相下采样<sup>[6]</sup>和编码预处理, 生成5个描述, 并将其存入SDRAM; 编码控制模块, 控制两个DSP以图组(Group of Pictures, GOP)为单位, 轮换编码各个描述。DSP中的主控程序将接收到的视频信号一帧帧地传送给DSP内部压缩算法。当需要对原始数字视频信号或者是压缩过程中需要继续使用的已压缩过的信号进行缓存时, 由DSP内部压缩算法将其存储在外部SDRAM上的指定缓冲区内。最终DSP压缩算法输出的压缩过的信号由DSP主控程序负责直接通过PCI总线连续地传送至主机。

### 2 预处理器的设计和实现

空域精细可扩展预处理器的硬件框图如图2所示, 主要包括3个模块: 下采样模块, 编码预处理模块和输出控制模块。图中缓存RAM的容量等于两帧图像的大小, 分成前后两个半区(frame\_1和frame\_2), 用来缓存帧的子层数据。

**基金项目:** 国家自然科学基金资助项目(69973007)

**作者简介:** 乔保军(1975-), 男, 讲师、博士生, 主研方向: ASIC设计方法, 嵌入式系统; 石峰, 教授、博导

**收稿日期:** 2006-01-05 E-mail: qbj@henu.edu.cn

采样模块向前半区写数据时, 编码预处理模块可读取后半区已写入的帧, 这种两帧数据并行加工处理的方式可以大大提高整个编码系统的运行效率。

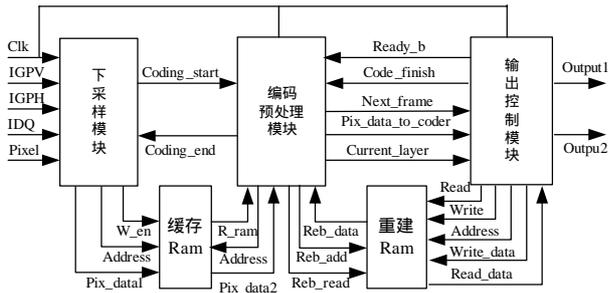


图2 预处理器硬件框图

### 2.1 下采样模块

该模块的主要功能是：从视频 A/D 和格式转换模块接收视频数据, 应用多相下采样算法将一个完整帧分解成 B1、B2、E1、E2 和 E3 5 个子图像, 并将这 5 个子图像写入缓存 RAM; 当所有数据都写入 RAM 后, 向 Coding\_start 信号线上发送一个脉冲启动编码预处理模块。下采样模块的主要输入除了系统时钟信号(Clk)和像素数据(Pixel)之外, 还有 3 控制信号: 帧有效(IGPV), 行有效(IGPH)和像素有效(IDQ)。

多相下采样的过程如图 3 所示, 图中以一个 8×8 的矩阵表示一个原始帧。

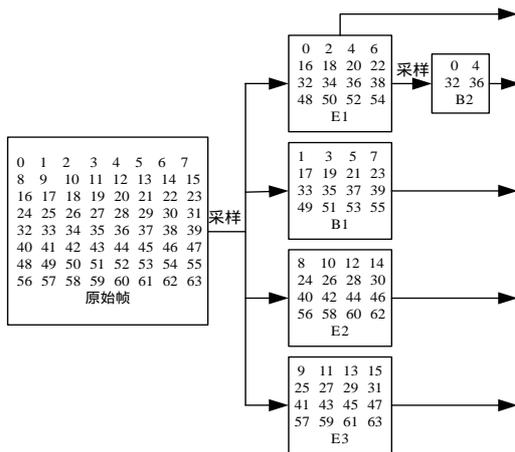


图3 下采样示例图

为了减少存储空间和提高实时处理的性能, 该模块对帧图像数据进行下采样处理时, 并不先将整个帧写入缓存 RAM 中, 然后在缓存 RAM 中对帧进行采样操作。该模块在接收像素数据的同时, 按照下采样算法的思想, 根据像素在帧中所处位置, 确定出像素所属的层, 将像素数据按照不同层号, 分别存放到 RAM 中对应的位置, 实现对帧数据的下采样。

下采样和编码预处理模块之间的关系是典型的生产者和消费者的关系, 这两个模块对缓存 RAM 的操作需要进行同步控制。下采样模块向缓存 RAM 中写入像素数据的前提是, 在缓存 RAM 中至少有一个空闲半区; 编码预处理模块从缓存 RAM 中读取数据的前提则是, 在缓存 RAM 中至少有一个半区已经存放有帧数据。系统初始时, 缓存 RAM 中的两个半区都是空闲, 下采样模块可以向缓存 RAM 中写入数据。如果两个半区都不空闲, 下采样模块必须等待, 直到编码预处理模块读取处理完一帧后, 它能够继续像素数据的写入操作。当缓存 RAM 的两个半区都空时, 编码预处理模块将处于待等状态, 只有下采样模块写入一帧后, 它能够继续

进行读取操作。

下采样和编码预处理两个模块之间的同步主要是通过 Coding\_start 和 Coding\_end 两个信号量和两个模块内部各自的状态量实现。当下采样模块生产完一帧后, 通过 Coding\_start 信号线发送一个脉冲以启动编码预处理模块; 在每一个时钟的上升沿, 下采样模块检测信号 Coding\_end, 如果 coding\_end 是高电平, 说明编码预处理模块从缓存 RAM 中消费完一帧。

在缓存 RAM 两个半区都满的情况下, 可能会出现丢帧现象, 但只要系统保证编码预处理模块的消费能力能够匹配下采样模块的生产能力, 就能够解决这个问题。

### 2.2 编码预处理模块

该模块主要的功能是从缓存 RAM 中读取一帧的像素数据, 对于不同子层的数据进行不同的处理, 得到 B1 和 B2 基本描述层和 E1'、E2' 和 E3' 增强层这 5 个最终描述码流, 并将这些描述码流传给输出控制模块。B1 和 B2 的数据将被直接传送给输出控制模块, 输出控制模块对 B1 和 B2 进行重建, 并将重建结果缓存到重建 RAM 中; 对于 E2 和 E3 的像素数据, 需要和从重建 RAM 中读取的 B1 重建数据进行相减运算, 得到 E2' 和 E3' 增强描述层; 对于 E1 的数据, 需要和 B2 重建数据插值之后的结果进行相减运算, 得到 E1' 增强层。图 4 给出子层数据插值和相减操作的示例。图 4 用 B1 和 B2 子层本身的数据作为它们的重建数据 B1\_R 和 B2\_R。

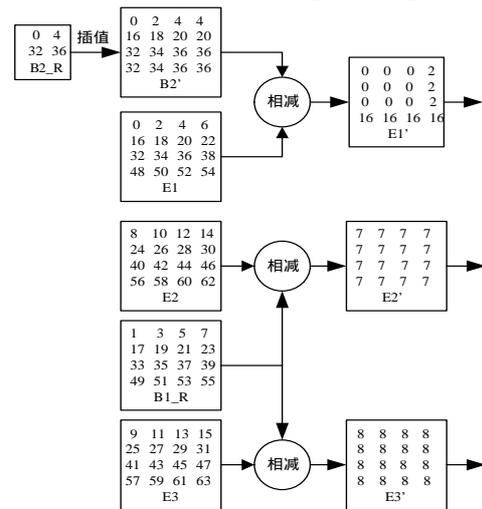


图4 插值和相减操作示例图

在最后得到的原始帧的 5 个描述码流中, B1 和 B2 能够独立解码, B1 重建图像分辨率为原始图像的 1/4, B2 重建图像分辨率为原始图像的 1/16; E1'、E2' 和 E3' 是预测残差, 不能独立解码。该精细可扩展编码算法的目的是: 减少带宽占用量; 提高传输可靠性。

当编码预处理模块处理完一帧后, 通过 Coding\_end 信号线向下采样模块发送一个脉冲信号, 通知该模块在 RAM 中有一个空闲半区可以使用。

编码预处理和输出控制模块之间也存在同步关系, 只有当输出控制模块处理完一帧数据后, 编码预处理模块才能开始新的一帧数据输出, 另外也只有当输出控制模块对当前帧的 B1 和 B2 子层重建及插值运算结束后, 预处理模块才能从重建 RAM 中读取相应的重建数据, 进行相减操作。信号线 Ready\_b 是用来启动重建数据的读取, 信号线 Code\_finish 是用来启动下一帧的处理。

### 2.3 输出控制模块

该模块主要的功能有：接收编码预处理模块传送来的像素数据，生成 B1 和 B2 的重建数据，并对 B2 重建数据进行插值操作，将这些重建及插值结果缓存到重建 RAM 中；将 B1 和 B2 作为基本层描述码流输出给图 1 中的 SDRAM(1)；从编码预处理模块接收 E1'、E2' 和 E3' 作为增强子层，输出给 SDRAM(2)；通知图 1 中的编码控制模块，使其控制两个 DSP 编码帧缓存 SDRAM 中的数据。输出控制模块内部充分利用数据操作的并行性，在接收 B2 的同时对 B1 进行重建操作，重建插值 B2 子层和接收 E2' 并行执行。这种操作并行可以大大提高帧图像编码预处理的性能。

### 3 仿真及 FPGA 实现结果

为了检验该编码预处理器的正确性，我们通过 ModelSim 进行了行为级功能和时序仿真。图 5 为时序仿真结果。

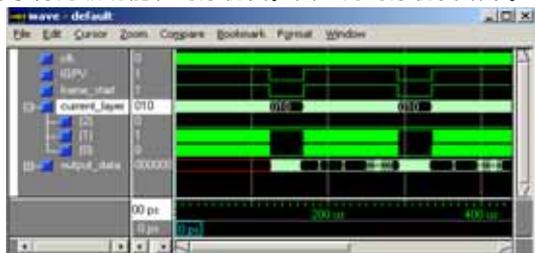


图 5 时序仿真结果

为验证算法的正确性，用 HDL Bench 生成测试激励，模拟一帧大小为  $352 \times 288$  的 24 位彩色图像输入，数据值从 0,0,0 到 255,255,255 循环递增。另外，在仿真验证时，将输出 Output1 和 Output2 合并成一个输出 Output\_data。时序仿真结果与理论计算结果一致。

该编码预处理器用 VHDL 代码在 Xilinx 的 ISE6.1 环境下设计完成，通过综合和布局布线，并在 Virtex2 XC2V500 的 FPGA 上完成调试。整个设计的面积占用率为 46%；管脚占用率为 87%；最高时钟频率约 39.8MHz；最小周期 25ns。在这样的情况下，完成一幅串行输入的  $352 \times 288$  的 24 位彩色图像需要约 60ms，即每秒钟可处理 15 帧。

### 4 结语

因其灵活性与并行性，FPGA 逐渐成为 DSP 进行视频信号处理不可或缺的协处理器。本文介绍的用 FPGA 实现空域精细可扩展编码预处理器的实例，已通过下载验证，结果表明该设计是正确有效的。今后的工作主要是，从提高数据操作的并行度出发，构造新的预处理器结构。

#### 参考文献

- 1 郭天天, 张志勇, 卢焕章. 快速浮点加法器的 FPGA 实现[J]. 计算机工程, 2005, 31(16): 202-204.
- 2 Waitian T, Zakhor A. Video Multicast Using Layered FEC and Scalable Compression[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11(3): 373-386.
- 3 Tham C H, Jiang Y M, Gan Y S. Layered Coding for a Scalable Video Delivery System[C]. Proc. of IEEE/EURASIP Packet Video, Nantes, France, 2003.
- 4 Goyal V K. Multiple Description Coding: Compression Meets the Networks[J]. IEEE Signal Processing Magazine, 2001, 18(5): 74-93.
- 5 王一拙, 石峰, 宋红. 空域精细可扩展编码算法的研究[J]. 北京理工大学学报, 2005, 25(3): 239-243.
- 6 Franchi N, Fumagalli M, Lancini R. Flexible Redundancy Insertion in a Polyphase Down Sampling Multiple Description Image Coding[C]. Proc. of Multimedia and Expo(ICME '02), New York, 2002: 605-608.

(上接第 179 页)

所示。接下来，我们用 Canny 算子和 Sobel 算子也对 3 幅原始图像进行边沿检测，其结果分别如图 3(g)~图 3(l)所示。

经比较不难发现：Canny 算子的检测结果细节部分过多；而 Sobel 算子则是检测边沿不完整。Canny 算子和 Sobel 算子中的参数同样需要人为设定，尽管经过足够多次的试验也能获得好的边沿检测结果，但在 Matlab6.5 中，这些参数的范围是从 0~1.0 之间选取，不同的图像需要不同的参数组合，并且检测结果对参数的选定极其敏感。比较改进的蚂蚁算法而言，改进蚂蚁算法中的参数设定相对简单，经过少数试验就可找到产生较好检测结果的初始参数。由蚂蚁算法的检测结果，细节部分较为适中，用虚点线较为清晰地勾勒出甲状腺结节的边沿，较好地反映了甲状腺结节的边沿特征，从而易于计算获得甲状腺结节面积等相关特征。由于甲状腺结节的诊断不仅依赖这些沿特征，而且还需观察结节的内部特征<sup>[2]</sup>。因此，我们结合初始超声图像及蚂蚁算法结果，仅将甲状腺结节边沿用虚点白线勾勒出，获得如图 4 所示的结果，使超声图像观察者能更好地对甲状腺结节进行诊断。



图 4 用改进的蚂蚁算法对图像仅作白线勾勒显示的结果

### 4 结论

由上面的分析，证明该改进的蚂蚁算法对于甲状腺结节

超声图像的边沿提取是有效的，它可以清晰地搜索出甲状腺结节的边沿，较好地反映出结节的边沿特征。在今后的工作中，我们将尝试改进算法中阈值的设定方法，将参数人为设定改为计算机求解最优阈值的选定，相信分割结果会有进一步的改进。

#### 参考文献

- 1 王允, 李普海. 结节性甲状腺肿声图像的特征分析[J]. 中国临床医学影像, 2000, 11(2): 129-130.
- 2 Hermus A R, Huysmans D A. Treatment of Benign Nodular Thyroid Disease[J]. The New England Journal of Medicine, 1998, 338(20): 1438-1447.
- 3 王怡, 唐天雪. 甲状腺超声图像特征[J]. 中国超声医学, 1998, 14(11): 19-20.
- 4 Liang L R, Looney C G. Competitive Fuzzy Edge Detection[J]. Applied Soft Computing, 2003, 3(2): 123-137.
- 5 韩彦芳, 施鹏飞. 基于蚁群算法的图像分割方法[J]. 计算机工程与应用, 2004, 40(18): 5-7.
- 6 Bonabeau E, Dorigo M, Theraulaz G. Swarm Intelligence: From Natural to Artificial Systems[M]. New York: Oxford University Press, 1999.
- 7 Shao F, Ling K V, Wu R Y, et al. Prostate Boundary Detection from Ultrasonographic Images[J]. J. of Ultrasound Med., 2003, 22(6): 605-623.
- 8 赵树魁, 李德玉. 超声医学图像滤波算法研究发展[J]. 生物医学工程, 2001, 18(1): 145-148.